DOCUMENT RESUME

ED 098 934 IR 001 310

| | |
|---|---|
| TITLE | Solo Works; Newsletter No. 27. |
| INSTITUTION | Pittsburgh Univ., Pa. Project Solo. |
| SPONS AGENCY | National Science Foundation, Washington, D.C. |
| PUB DATE | May 74 |
| NOTE | 7p. |
| EDRS PRICE | MF-$0.75 HC-$1.50 PLUS POSTAGE |
| DESCRIPTORS | Computer Programs; Computers; Computer Science; *Computer Science Education; Educational Technology; *Newsletters; Secondary Education |
| IDENTIFIERS | *Project SOLO; University of Pittsburgh |

ABSTRACT

A newsletter for the SOLO Project at the University of Pittsburgh contains short problems and solutions of a general nature that arise in computer technology for education and computer science education. The recent and planned activities of the SOLO Project are also listed. Solo Works Newsletter No. 25 is a poster and No. 26 is a journal reprint; therefore neither is available through ERIC. (WH)

ED 098934



# SOLO WORKS

Project Solo,
University of
Pittsburgh,
Pittsburgh, Pa.
15260 (412) 624-6461

Newsletter #27 May 1974

Project 5020
Dept. of Computer Science
University of Pittsburgh
Pittsburgh, Pa. 15213

Project FOLO
70 Cathedrial of Learning
University of Pennsylvania
Pittsburgh, Pa. 15260

University of Pittsburgh
Project Soul
812 Cathedral of Learning
Pittsburgh, Pa.

PROJECT SOLE
[illegible] OF COMPUTER SCIENCE
[illegible] PA. 15260

SOLO, PROJECT
DEPARTMENT OF COMPUTER SCIENCE
UNIVERSITY OF PITTSBURGH
PITTSBURGH, PA. 15213

Dr. Thomas Dwyer, Director, Project SOLD
University of Pittsburgh
Department of Computer Science
Pittsburgh, Pa. 15213

Project SOLD
University of Pittsburgh
Pittsburgh, Pennsylvania 15213

SPECIAL DELIVERY

U.S. DEPARTMENT OF HEALTH
EDUCATION & WELFARE
NATIONAL INSTITUTE OF
EDUCATION
THIS DOCUMENT HAS BEEN REPRODUCED EXACTLY AS RECEIVED FROM THE PERSON OR ORGANIZATION ORIGINATING IT. POINTS OF VIEW OR OPINIONS STATED DO NOT NECESSARILY REPRESENT OFFICIAL NATIONAL INSTITUTE OF EDUCATION POSITION OR POLICY

Dr. Tom Dwyer
University of Pittsburgh
Cathedral of Learning
PROJECT SOLD
Room 701
Pittsburgh, Pennsylvania

Project Solo Capt
812 Cathedral of Learning
Pittsburgh, Pa. 15206

## THANK YOU FOR THE LETTERS

We asked for it--and we got it. So many letters that we can't begin to answer them individually in the manner they deserve. But we do guarantee that your comments and suggestions are having an important influence on our work at the Soloworks lab.

One of the most important things this correspondence is telling us, both directly and indirectly, is that the real world of teachers is characterized by a sensitivity, creativity, and intelligence that fairly cry out for recognition as a resource to be tapped and built upon. Yet is is a resource that is being by-passed by all too many programs in educational technology. Not just discounted or ignored, but deliberately by-passed.

By-passing the human teacher element when first applying technology to education might be defended as the expedient thing to do. It gives the ed-tech R&D lab a neater and tidier world in which to work, one where the technology-age virtues of "replicability" and "cost-effectiveness" can be better controlled. But the cost of such expediency is high; by-passing teachers means by-passing a style of education (predicated very much on technology) that is unprecedented in its possibilities.

This claim is not made lightly; it comes out of real and compelling experiences both in Project Solo schools, and in the Soloworks lab. It's based on observations in classrooms like that of Irwin Hoffman In Denver. And we've seen it at work in other labs.

We recently visited the Logo Project at MIT where Seymour Papert's group is exploring the use of computers and other technology

ER 001 310

to enhance education. Enhance is certainly the right word (as opposed to automate, manage, or mass-produce). In fact the deepest impression of the day was made on us by a young teacher working with some very ordinary looking kids, and the manner is which they reacted to the unscheduled tipping of a can of pop into an expensive terminal. Their handling of the situation was problem-solving at its finest. Nary a rant nor a rave ensued; instead there was a quick mopping, followed by analysis and planning to handle future perturbations of this kind, all in a friendly spirit that one would hope to see throughout society some day. For more on this subject see item #29 on page 5.

## RSTS AND THE MAGIC MINI

Mini-RSTS is a computer system that--unbelievable as it may sound--is even better than the manufacturer claims. RSTS means resource time-sharing, which in turn means a very nifty operating system that allows terminals, discs, and all kinds of other I/O hardware to access each other and a PDP 11/40 minicomputer. The high level language on the system is BASIC-PLUS which is also very nifty.

As you may gather, Soloworks is now using a mini-RSTS system, and with great enthusiasm. Examples to show why will be coming your way in future newsletters (a brief one appears on the last page of this newsletter). In the meantime, we should mention that the ability to custom-tailor the system is very fascinating to students. They've modified ours so that is contains a program which sleeps for fifteen minutes at a time. When it wakes up, the Westminister Chimes theme and the correct time are automatically played on a small pipe organ. Now that's resource sharing.

## MORE ON BASIC-PLUS

Sometime back we received an invitation to join the Society to Help Abolish Fortran in Teaching. It was easy to sympathize with that group; FORTRAN was a good idea when first introduced, but its topsy-turvy growth has resulted in arbitrary detail that distracts from a human-oriented approach to problem exploration (and the kind of learning this implies). The philosophy of languages like JOSS and BASIC was much more on target.

Of these two, BASIC has received the biggest push for acceptance, and it is now in use around the world. It's main drawback is that the original specification was too basic. Students of all ages very quickly move on to a level that cries out for better and more intricate ways to control the computer. The languages NEWBASIC (Com-Share Co.), and BASIC-PLUS (Digital Equipment Corporation) are excellent solutions to this problem. Our experience has been that sophisticated software of this calibre makes a big difference in what kids do with computers; it's like having some very heavy shackles on one's imagination melted away.

CURRICULUM PLUS

The proof of these statements will of course be in the pudding of extended curriculum ideas. Such curriculum ideas will also serve as benchmarks for the evaluation of educational software. Most vendors now label their version of BASIC as being "extended-BASIC". This can be pretty deceptive, and the only way to separate the wheat from the chaff is to start with extended educational ideas, and then see if the software can handle them. To this end, we're starting to assemble some educational benchmarks that illustrate the possibilities. This work, which is coming out of the Soloworks Computer and Modelling labs, is currently organized into the following categories:

## ✪ COMPUTER PROGRAMS FOR KIDS BY KIDS

What happens when you put kids and BASIC-PLUS together, and simply say GO? We've been trying this with students aged 9 to 18. The results are intriguing. They suggest that there is a bad mismatch between much school practice and human nature at these age levels. They also suggest that a learning theory which builds on the more subtle nuances in human nature will be more fruitful than one based on behavior modification. To this end, we are developing a learning model based on those activities that we've found to work extraordinarily well with students. Our goal is to learn how to fine-tune educational environments to support something we call "resonant learning". Jargon? Maybe. Does it work? Like a charm (so far).

## ✪ MATHEMATICS, COMPUTATION, AND THE MYSTERIOUS WORLD OF NUMBERS

Technically this will be a series that brings together high school math and modern numerical analysis. Without computers this would be an impossible goal. With computers, and a language that permits things like double precision, it looks like a winner. Kids of all ages are fascinated with programs that generate sequences of 15-digit numbers which zoom in on the answers to problems that defy textbook methods. Here's an example of such a problem (a solution is given on page 6).

Starship Gamma has reached the point in its journey where the Captain is authorized to open the Kryptonium safe containing mission orders. The combination to the safe consists of ten 10-digit numbers as shown below--except...Well, read for yourself.

MEMO TO CAPTAIN, STARSHIP GAMMA:

(A) THERE IS ONE INCORRECT DIGIT IN ONE OF THE NUMBERS IN YOUR COMBINATION ⟶

(B) THE NUMBERS WERE GENERATED BY A CONTINOUS FUNCTION THAT IS APPROXIMATED BY A 4TH DEGREE POLYNOMIAL.

(C) IF YOU ENTER THE INCORRECT DIGIT
*S*P*L*A*T* !!!!

(D) IT IS THEREFORE HIGHLY RECOMMENDED THAT YOU USE YOUR COMPUTER TO FIND THE BAD DIGIT.

0.5226872289
0.5235394920
0.5243912314
0.5252424466
0.5260931364
0.5269433002
0.5277929370
0.5286420461
0.5294906264
0.5303386773

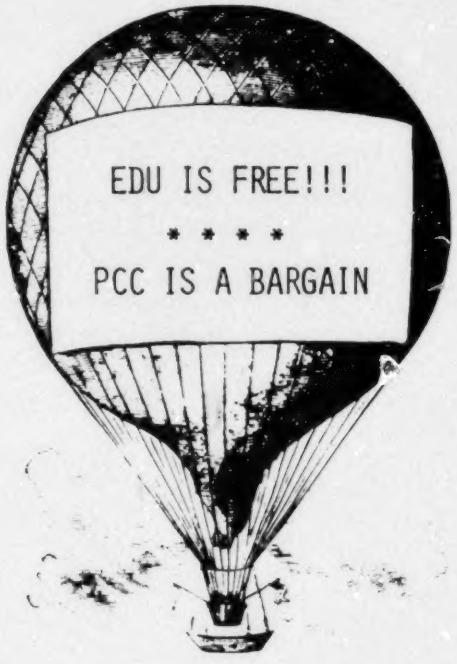
## GAMES, FANTASY, MODELLING, AND SIMULATION

If you have a game club in your school, or a good version of Star Trek on your computer, you'll know why we've identified these ideas as "resonant" phenomena. Our intention is to build on this interest, and create extensible versions of such game-like programs. This means that the games should be based on clearly defined structures, either student-invented or classical (e.g. Euclidean Geometry, abstract vector spaces, directed graphs, etc.). The possibility of learning through experience with concretized abstractions looks very promising.

## COMPUTER SCIENCE AND COMPUTER LITERACY

More and more schools are introducing a course in computer science into the curriculum. There is also much discussion about whether "computer literacy" shouldn't be a part of everyone's education. We feel that the best way to learn about computer science is to work directly with the major ideas of computer science. The curriculum units in this series will therefore be lab-oriented, with suggestions for hands-on computing projects related to selected readings. Topics will range from the theoretical (e.g. artifical intelligence), to the very utilitarian (e.g. how to design and operate a student-record system for your school's computer). For more on student-record systems, see the enclosed questionnaire.

NOTE: Please don't write us for any of these units yet. We'll let you know if and when they are printed for general use.




## * * * * MINI-BILLBOARD * * * *

- For a free subscription to EDU write to:

  Mrs. Sally Bower
  Educational Products Group
  146 Main Street
  Maynard, MA 01754

- We recently went over our back issues of the People's Computer Company Newsletters, and found a wealth of good ideas buried therein. We recommend both a new subscription ($5.) and purchase of back issues ($2.). Send orders to:

  People's Computer Company
  P. O. Box 310
  Menlo Park, CA 94025

- The Computer Resource Book--Algebra will be available from Houghton-Mifflin, One Beacon Street, Boston, MA 02107 in December (they've promised).

## PROJECT SOLO & SOLOWORKS NEWSLETTERS

Project Solo newsletters are out of print. Some bona-fide reprints of PS modules are available from DEC, and some modified PS modules can be purchased from HP. The switch to Soloworks took place with newsletter #23. We now print 1,000 Soloworks Newsletters, and (unless there's an unexpected $$$ bonanza from the blue) will not be able to reprint. But a number of our publications will be reprinted elsewhere, and we'll keep you informed. Here's the box score to date:

| | Soloworks Newsletter | Contents | Can Also be Found in: |
|---|---|---|---|
| Mailed Jan. 1974 | #23 November 1973 | Description of the Soloworks Project | EDU #12 and PCC May 1974 |
| | #24 December 1973 | Content of the Soloworks Labs | EDU #12 and PCC May 1974 |
| Mailed Oct. 1974 | #25 March 1974 | Soloworks Lab Poster | Collector's item |
| | #26 April 1974 | Heuristic Strategies for Using Computers to Enrich Education | Intl. Jnl. Man-Machine Studies, Vol. 6, 1974, pg. 137-154. |
| | #27 May 1974 | You're reading it right now! | |
| Not Mailed | #28 June 1974 | The Significance of Solo-Mode Computing for Curriculum Design | EDU #13, (September 1974) |
| | #29 July 1974 | Some Thoughts on the Importance of Greatness in Teaching | To appear in a special SIGCUE annual on computers and teacher education. |
| | #30 August 1974 | Computers and the Romantic View of Education | Technological Horizons in Education, Vol. 1, No. 3. Write: P.O. Box 992, Acton, MA 01720 (Free) |

## MEANWHILE, BACK AT STARSHIP GAMMA

```
10 DIM A(20,5): MAT A=ZER
15 READ A(I,1) FOR I=1 TO 19 STEP 2
20 FOR J=2 TO 5: FOR I=J TO 20-J STEP 2
25 LET  A(I,J)=A(I+1,J-1)-A(I-1,J-1)
30 NEXT I: NEXT J
35 LET D$="#.##########  "
40 LET S$="              "
45 FOR I=1 TO 19: FOR J=1 TO 5
50 IF A(I,J)=0 THEN PRINT S$;
              ELSE PRINT USING D$, A(I,J);
55 NEXT J: PRINT: NEXT I
60 DATA .5226872289, .5235394920, .5243912314
65 DATA .5252424466, .5260931364, .5269433012
70 DATA .5277929370, .5286420461, .5294906264
75 DATA .5303386773
```

Here's a BASIC-PLUS program to produce a difference table for the Starship Gamma problem. The format in line 35 is used to produce numbers with ten decimal places. (BASIC-PLUS can go up to fifteen decimal places.)

The output of this program is the difference table shown on the next page.

Solution to the "Bad Digit" Problem (or how the Captain of Starship Gamma opened his safe without triggering the SPLAT annihilator beam).

The Captain remembered two things from his Cosmic-Algebra-I class:

1. A polynomial of degree n will have zeros for its n+1 differences. Here's an example for the polynomial $P(x)=X^2+10$:

| X | P(X) | 1st Differences | 2nd Differences | 3rd Differences |
|---|---|---|---|---|
| 1 | 11 | | | |
| | | 3 | | |
| 2 | 14 | | 2 | |
| | | 5 | | 0 |
| 3 | 19 | | 2 | |
| | | 7 | | 0 |
| 4 | 26 | | 2 | |
| | | 9 | | |
| 5 | 35 | | | |

9=35-26 2=9-7 0=2-2 etc.

For a second degree polynomial, the third differences are zero.

Rule: Each "difference" is calculated by subtracting the two entries to its left.

2. Any error E in P(x) will propogate through the difference table with a Pascal triangle pattern as follows:

| | | | | | |
|---|---|---|---|---|---|
| | | | | +E | etc. |
| | | | +E | | etc. |
| | | +E | | -4E | |
| | +E | | -3E | | etc. |
| +E | | -2E | | +6E | |
| | -E | | +3E | | etc. |
| | | +E | | -4E | |
| | | | -E | | etc. |
| | | | | +E | |

To convince yourself of this, make a difference table for $P(x)=X^4+10$ with X=0, 1, 2, . . . , 10. Put an error in P(5) by making it 636 instead of 635, and see what happens.

For our starship problem, the difference table looks like the following:

Notice the pattern 1, -4, 6, -4, 1 that appears in the ninth decimal place. This is a dead give-away that there was an error E = +1 in the ninth place of 0.5269433012.

| | | | | |
|---|---|---|---|---|
| 0.5226872289 | | | | |
| | 0.0008522631 | | | |
| 0.5235394920 | | -.0000005237 | | |
| | 0.0008517394 | | -.0000000005 | |
| 0.5243912314 | | -.0000005242 | | -.0000000007 |
| | 0.0008512152 | | -.0000000012 | |
| 0.5252424466 | | -.0000005254 | | 0.0000000016 |
| | 0.0008506898 | | 0.0000000004 | |
| 0.5260931364 | | -.0000005250 | | -.0000000044 |
| | 0.0008501648 | | -.0000000040 | |
| 0.5269433012 | | -.0000005290 | | 0.0000000063 |
| | 0.0008496358 | | 0.0000000023 | |
| 0.5277929370 | | -.0000005267 | | -.0000000044 |
| | 0.0008491091 | | -.0000000021 | |
| 0.5286420461 | | -.0000005288 | | 0.0000000015 |
| | 0.0008485803 | | -.0000000006 | |
| 0.5294906264 | | -.0000005294 | | |
| | 0.0008480509 | | | |
| 0.5303386773 | | | | |

Thus the sixth number of the combination should have been: 0.5269433002.